6

NASA TECHNICAL MEMORANDUM NASA TM-75678

(NASA-TM-75678) PROGRAMS FOR HIGH-SPEED FOURIER, MELLIN AND FOURIER-BESSEL TRANSFORMS (National Aeronautics and Space Administration) 19 p HC A02/MF A01 CSCL 09B N80-18749

Unclas
G3/61 47190

PROGRAMS FOR HIGH-SPEED FOURIER, MELLIN AND FOURIER-BESSEL TRANSFORMS

Dr. K. Tkhabisimov, A. S. Debabov, B. I. Kolosov,
D. A. Usikov

Translation of "Programmy Bystrykh Preobrazovaniy Fur'ye, Mellina i Fur'ye-Besselya", Academy of Sciences USSR, Institute of Space Research, Moscow, Report Pr-418, 1978, pp 1-17

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
Washington, D. C. 20546 September 1979

NOTICE

THIS DOCUMENT HAS BEEN REPRODUCED FROM THE BEST COPY FURNISHED US BY THE SPONSORING AGENCY. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE.

STANDARD TITLE PAGE

1. Report No. NASA TM-75678	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle PROGRAMS FOR HIGH-SPEED FOURIER, MELLIN AND AND FOURIER-BESSEL TRANSFORMS		5. Report Date September 1979
		6. Performing Organization Code
7. Author(s) D. K. Tkhabisimov, A. S. Debabov, B. I. Kolosov, D. A. Usikov		8. Performing Organization Report No.
		10. Work Unit No.
9. Performing Organization Name and Address SCITRAN Box 5456 Santa Barbara, CA 93108		11. Contract or Grant No. NASW-3198
		13. Type of Report and Period Covered Translation
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code

15. Supplementary Notes Translation of "Programmy Bystrykh Preobrazovaniy Fur'ye, Mellina i Fur'ye-Besselya", Academy of Sciences USSR, Institute of Space Research, Moscow, Report Pr-418 1978, pp. 1-17

16. Abstract

We present a description of program modules for performing one-dimensional and two-dimensional discrete Fourier transforms, Mellin and Fourier-Bessel transforms, and also programs for realizing the algebra of high-speed Fourier transforms on a computer. The programs developed can be used to perform numberical harmonic analysis of functions, to synthesize complex optical filters on a computer, modelling the holographic methods of processing images.

The programs are written in FORTRAN and are included in the library of programs of the video images processing unit "SOFI" in the Institute of Space Sciences of the USSR Academy of Sciences.

17. Key Words (Selected by Author(s))	18. Distribution Statement Unclassified - Unlimited

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22.
Unclassified	Unclassified	17	

TABLE OF CONTENTS

Preceding page blank

PROGRAMS FOR HIGH-SPEED FOURIER, MELLIN AND FOURIER-BESSEL TRANSFORMS

D. K. Tkhabisimov, A. S. Debabov, B. I. Kolosov, D. A. Usikov

We present a description of program modules for performing one-dimensional and two-dimensional discrete Fourier transforms, Mellin and Fourier-Bessel transforms, and also programs for realizing the algebra of high-speed Fourier transforms on a computer. The programs developed can be used to perform numerical harmonic analysis of functions, to synthesize complex optical filters on a computer, modelling the holographic methods of processing images.

The programs are written in FORTRAN and are included in the library of programs of the video images processing unit "SOFI" in the Institute of Space Sciences of the USSR Academy of Sciences.

INTRODUCTION

/3*

The need for numerical harmonic analysis arises when solving integral equations of convolution type. Here all the functions are given in discrete form. Let $f(n)$ be the file of the readings for the initial function $f(x)$ which is nonzero on the interval (a, b) of the real axis. A pair of discrete Fourier transforms (DFT) of the file $f(n)$ has the form:

$$\overset{*}{f}(m) = D_{mn} f(n) = \frac{1}{N} \sum_{n=0}^{N-1} e^{i \frac{2\pi}{N} mn} f(n) \tag{1}$$

and

$$f(n) = D^{-1}_{nm} \overset{*}{f}(m) = \sum_{m=0}^{N-1} e^{-i \frac{2\pi}{N} mn} \overset{*}{f}(m), \tag{2}$$

*Numbers in the margin indicate pagination of original foreign text.

where N is the dimension of the files f(n) and $f^*(m)$. By means of the transforms (1) and (2) we can compute continuous Fourier transforms and the coefficients of the Fourier series of the initial function [1]. The Mellin transform of the function $f(x)$ is defined by the formula [2]:

$$f^*(\omega) = M_{\omega,x} f(x) = \int_0^\infty x^{i\omega - 1} f(x)\, dx. \tag{3}$$

On making the substitution $x = e^t$ under the integral sign in (3), we obtain

$$f^*(\omega) = \int_{-\infty}^{\infty} e^{i\omega t} f(e^t) e^{2t} dt = \Phi_{\omega,t}\{f(e^t)\cdot e^{2t}\} \tag{4}$$

where $\Phi_{\omega,t}$ is the operator of the continuous Fourier transform. From (4) it is evident that the Mellin transform is carried out with the help of (1). The Fourier-Bessel function $f(x)$ has the form:

$$f_n^*(\tau) = B_{\tau,x}^{n} f(x) = \int_0^\infty J_n(x\tau) f(x) x\, dx. \tag{5}$$ /4

In [1] a procedure is shown for computing (5) with the help of the transforms (1)-(3). Applying the Mellin transform (3) to (5), we obtain

$$(n>0)$$

$$M_{\omega,\tau} f_n^*(\tau) = \bar{M}_{\omega,\rho} f(\rho)\rho^2 M_{\omega,\rho} J_n(\rho)$$

or

$$M_{\omega,\tau} f_n^*(\tau) = \Phi_{\omega,t}^{-1}\{f(e^t)e^{2t}\} W_n(\omega), \tag{6}$$

where

$$W_n(\omega) = \frac{2^{i\omega}}{n - i\omega} \frac{\Gamma\left(\frac{n+i\omega}{2}\right)}{\Gamma\left(\frac{n-i\omega}{2}\right)}. \tag{7}$$

The coefficient $f_n^*(\tau)$ is computed at the points where $\tau = e^u$ $(-\infty < u < \infty)$. The case n = 0 reduces to the case n = 1 if we note that

$$\int_0^\infty J_0(\rho x) f(\rho)\rho\, d\rho = \tau \int_0^\infty \mathcal{F}(\rho) J_1(\rho\tau)\rho\, d\rho,$$

where

$$\mathcal{F}(\rho) = \frac{1}{\rho}\int_0^{\rho} f(s)\, s\, ds$$

To calculate (7) we can use the representation [2]:

$$(a + i\omega) = \int_{-\infty}^{\infty} e^{-e^{t}+at}\, e^{i\omega t}\, dt = \Phi_{\omega,t}\left(e^{-e^{t}+at}\right), \tag{8}$$

where $a \leq 1$. When $a > 1$ we make use of the recursion formula:

$$\Gamma(z+1) = z\,\Gamma(z)$$

Bessel's functions are computed by means of (1), since

$$J_n(\rho) = \frac{1}{2\pi}\int_0^{2\pi} e^{in\theta}\, e^{-i\rho\sin\theta}\, d\theta \tag{9}$$

*

In the present paper, programs are described which realize the transforms (1), (2), (3) and (5). A description is also given of programs for the algebra of high-speed Fourier transforms, programs for calculating the Bessel and gamma functions, etc. The text of the programs appears in the appendix.

1. Program for the High-Speed Fourier Transform

1.1 The One-Dimensional Version

The programs DQFT1 and RQFT1 are designed to carry out direct and inverse DFT of one-dimensional files described in the user's program as a complex. In accordance with the principles of organization of the complex "SOFI" [4], access to the programs is accomplished as follows:

1) In the basic program the user defines

```
COMPLEX A(n)
COMMON FICT(395), N, IJ(3), A
N=n
```

*Illegible in original foreign text.

where A is a working one-dimensional file with the DFT program to be processed $(n = 2^{m})$;

2) The file to be processed is entered into the working file A and accesses one of the programs of the high-speed Fourier transforms, e.g., CALL DQFT1;

3) The Fourier transform of the initial file is placed in the file A; the error in the calculation does not exceed 10^{-8} when $n \leq 2^{10}$.

1.2 The Two-Dimensional Version

The direct and inverse DFT of two-dimensional files is performed with the help of the programs DQFT2 and RQFT2. Here not every variable undergoes (1) and (2). In the basic program it is necessary to write: /6

```
COMPLEX  A(n1,n2), B(n1,n2)
COMMON FICT(395), N, NN, TB(2), A, B
N=n1
NN=n1*n2
```

where A, B are the one-dimensional working files with the DFT program to be processed $(n_1 = 2^{m_1}; n_2 = 2^{m_2})$. Moreover, analogous to Sec. 1.1, a description of the programs used in the high-speed Fourier transform programs can be found in [3].

When formulating the basic program, the user must designate the library of the complex for which the card

```
// PAUSE ASSGN SYSRLB,X'19N' gurk 102
```

must be inserted after the card

```
// JOB NAME
```

2. The Fourier-Bessel Transform

The Fourier-Bessel coefficients (5) are computed by the program FTFB(F, NT, NB, W, T0, TM). Here F is the file for the readings of the initial function f(x) at the points $x = e^{t}$, NT is the number of readings on the interval $t_0 \leq t \leq t_{max}$, NB is the subscript of the Bessel function, W is the

file of the function $W_h(\omega)$, $T\emptyset = t_0$, TM = t_{max}. For access to the program FTFB, the user must write in the basic program

```
COMPLEX   F(m), A(m)
COMMON FICT(399), A
```

where m is the dimension of the file. The result is located in the file F.

The values of the function $W_h(w)$ are calculated by means of the program FILT(W, NT, NB, TT). Here W is the desired file of values of the function /7
$W_h(\omega)$. NT is the number of values, $NB = n$, $TT = t_{max} - t_0$, the function $W_h(\omega)$ is computed at the points $\omega_K = \frac{2\pi}{TT} K$; $K = 0, \ldots, NT-1$. In the program FILT, the gamma function is calculated with the help of series (cf. Sec. 3); however, it is also possible to make use of the representation (8). The programs FTFB and FILT are accessed by means of the operator CALL.

3. Calculating the Bessel and Gamma Functions

The program BESF(BF,TF,NF) is calculated by means of formula (9) NF of the Bessel functions at the point TF:

$$J_0(TF), J_1(TF), \ldots, J_{NF-1}(TF)$$

and the result is placed in the file BF(NF).

The program GAMMA1(G, NT, TØ, TM, AS) computes the function $\Gamma(AS + i\omega)$ at the points $\omega_K = \frac{2\pi}{TT} K$, $K = -\frac{NT}{2}, \ldots, \frac{NT}{2} - 1$, where $TT = TM - T\emptyset$, TM and TØ are the upper and lower integration limits in (8). The result is located in the file G(NT). Before accessing programs BESF and GAMMA1, it is necessary to describe:

```
COMPLEX  A(n)
COMMON  FICT(399), A
```

The program GAMMA(Z) calculates the gamma function at the point $Z = a + ib$ by means of the formula [5]:

$$\Gamma(z) = \lim_{n \to \infty} \frac{n^z}{z} \prod_{K=1}^{n} \frac{K}{z + K}.$$

The result is located in Z.

4. Auxiliary Programs

Below we give descriptions of programs whose use simplifies access to the high-speed Fourier transform programs and accelerates the process of formulating a basic program. All programs work with complex files. /8

1) DLINE(A,B,NX,NY,K,L)- the program selects line number K from file A(NX,NY) and when L = 0, copies the information from file A into file B(NY); when L = 1, it copies the information from file B(NY) into file A.

2) CLMN(A,B,NX,NY,K,L) -- the program selects column number K from file A(NX,NY); when L = 0, it copies the information from file A into file B(NX); when L = 1, it copies the information from file B(NX) into file A.

3) OUTCOM(A,NX,NY,K)- -- the program copies the information from the COMMON-block of element K in file C into file A(NX,NY).

4) TOCOM(A,NX,NY,K) -- the program copies the information into the COMMON-block of element K in file C from file A(NX,NY).

5) AMULT(A,B,S,NX,NY)-- the program multiplies out the two two-dimensional files A(NX,NY and B(NX,NY), component by component. The result is located in the file S(NX,NY).

6) TRNSP(A,B,NX,NY) -- the program transposes the matrix A(NX,NY). The result is located in the file B(NX,NY).

7) CONJG(A,B,NX,NY) -- the program forms the complex conjugate of the matrix A(NX,NY). The result is located in the file B(NX,NY).

8) POULR(A,HX HY,NX,NY,B,NT,NF,R0,RM)-this is a program for the linear interpolation of a function, given in Cartesian coordinates by the file of values A(NX,NY) on a network with sampling steps HX and HY along the axes OX and OY, respectively, into the polar exponential network (e^t, φ), where

$$x = e^t \cos\varphi, \quad y = e^t \sin\varphi, \quad -\infty < t \leq \infty, \quad 0 \leq \varphi \leq 2\pi.$$

Here $R0 = e^{t_0}$, $RM = e^{t_{max}}$, NR -- number of subdivisions of the network with respect /9

to the variable t, NF -- number of subdivisions of the network with respect to the variable φ. The result is located in the file B(NT,NF).

9) $REVOL(F, N\emptyset, N1, V, NX, NY, AL)$ -- is a program of linear interpolation of the values of a function given in the Cartesian file $F(N\emptyset, N1)$ into the Cartesian network rotated through the angle AL relative to the origin of coordinates. The origin of coordinates corresponds to the point $\left(\frac{NX}{2}+1, \frac{NY}{2}+1\right)$.

NX and NY are the numbers of the nodes of the new network with respect to each variable (the sampling step is not changed); V(NX,NY) is the file of values of the initial function in the new network.

10) $CORF(F, V, S, NX, NY)$ -- the program calculates the correlation function of the two images in the plane given by the files F(NX,NY) and V(NX,NY) of the form:

$$S(y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x-y)\, v(x)\, dx, \qquad (10)$$

where x, y are vectors with components x_1, x_2 and y_1, y_2, respectively. In the result, the file S(NX,NY) of the readings of the correlation function is obtained.

CONCLUSION

In the present paper the programs described are included as object modules in the complex "SOFI" and are intended for various problems in spectral processing; recognition of images, refinement of images, the photographing of smog, the effects of blurring, for various statistical and physical problems required when applying the algorithms for the Fourier, Mellin and Fourier-Bessel transforms.

The authors express their deep appreciation to V. G. Zolotykhin for useful discussions.

REFERENCES

1. D. A. Usikov. Primeneniye abstraktnogo gapmonicheskogo analiza dlya bystrogo racpoznavaniya izobrazheniy (The Application of Abstract Harmonic Analysis to High-Speed Identification of Images). Preprint The Space Sciences Institute of the USSR Academy of Sciences, Pr-335, Moscow, 1977.

2. N. Ya. Vilenkin. Spetsial'nyye funktsii i teoriya predstavleniy grupp (Special Functions and the Theory of Groups). Nauka, Moscow, 1965.

3. A. S. Debabov, D. A. Usikov. Spektral'nyye preobrazovaniya v komplekce obrabotki fotograficheskikh izobrazheniy "SOFI" iz YeS-EVM (Spectral Transforms in the Complex for Processing Images, "SOFI" on a YeS [Unified Series[Computer), Part 1, Preprint, The Space Sciences Institute of the USSR Academy of Sciences, Pr-383, Moscow, 1977.

4. V. G. Zolotykhin, V. I. Kolosov, D. A. Usikov, et al. Opisaniye sistemy program matematicheskoy obrabotki na EVM Yedinoy Serii (YeS) fotograficheskikh izobrazheniy Zemli, poluchennykh s kosmicheskikh apparatov (kompleks "SOFI") (Description of a System of Programs for Mathematically Processing on a Unified Series (YeS) Computer Photographic Images of the Earth Obtained from Space Vehicles (The Complex "SOFI")). Preprint, The Space Sciences Institute of the USSR Academy of Sciences, Pr-336, Moscow 1977.

5. Yanko, Ye., Emde F., Aosh F. Special'nyye funktsii (Special Functions). Nauka Press, Moscow 1964.

APPENDIX

DQFT1

```
      SUBROUTINE DQFT1
      COMMON FICT(394),NC,N,NN,NA,NK,C(1)
      NN=N
      N2=N/2
      N3=N2-1
      NK=1
      NA=NK+2*N
      DO 1 I=1,N
      C(NA+I-1)=C(NK+2*I-2)
    1 C(NA+N+I-1)=C(NK+2*I-1)
      NK2=NK
      NK=NA+2*N-1
      NC=0
      CALL FFTD
      NC=3
      CALL FFTD
      NK=NK2
      C(NK)=C(NA)
      C(NK+1)=C(NA+N)
      C(NK+N)=C(NA+N2)
      C(NK+N+1)=C(NA+N+N2)
      DO 2 I=1,N3
      NI=N-I
      C(NK+2*I)=C(NA+I)-C(NA+N+NI)
      C(NK+2*NI)=C(NA+I)+C(NA+N+NI)
      C(NK+2*I+1)=C(NA+NI)+C(NA+N+I)
    2 C(NK+2*NI+1)=C(NA+N+I)-C(NA+NI)
      RETURN
      END
```

RQFT1

```
      SUBROUTINE RQFT1
      COMMON FICT(394),NC,N,NN,NA,NK,C(1)
      NN=N
      N2=N/2
      N3=N2-1
      NK=1
      NA=NK+2*N
      C(NA)=C(NK)
      C(NA+N)=C(NK+1)
      C(NA+N2)=C(NK+N)
      C(NA+N+N2)=C(NK+N+1)
      DO 1 I=1,N3
      NI=N-I
      C(NA+I)=0.5*(C(NK+2*I)+C(NK+2*NI))
      C(NA+NI)=0.5*(C(NK+2*I+1)-C(NK+2*NI+1))
      C(NA+N+I)=0.5*(C(NK+2*I+1)+C(NK+2*NI+1))
    1 C(NA+N+NI)=0.5*(C(NK+2*NI)-C(NK+2*I))
      NK2=NK
      NK=NA+2*N-1
      NC=-3
      CALL FFTD
      NC=0
      CALL FFTD
      NK=NK2
      DO 2 I=1,N
      C(NK+2*I-2)=C(NA+I-1)
    2 C(NK+2*I-1)=C(NA+N+I-1)
      RETURN
      END
```

DQFT2

```
      SUBROUTINE DQFT2
      COMMON FICT(394),NC,N,NN,NA,NK,C(1)
      NK=1
      NA=1+3*NN
      NC=6
      CALL INRECO
      NA=NA-NN
      NC=7
      CALL INRECO
      NK=NA+2*NN-1
      NC=0
      CALL FFTD
      NC=2
      CALL FFTD
      XN=1./FLOAT(NN)
      DO 1 I=NA,NK
    1 C(I)=C(I)*XN
      NK=1
      NC=1
      CALL INRECO
      NC=2
      NA=NA+NN
      CALL INRECO
      RETURN
      END
```

RQFT2

```
      SUBROUTINE RQFT2
      COMMON FICT(394),NC,N,NN,NA,NK,C(1)
      NK=1
      NA=1+3*NN
      NC=4
      CALL INRECO
      NA=NA-NN
      NC=3
      CALL INRECO
      NC=-2
      NK=NA+2*NN-1
      CALL FFTD
      NC=0
      CALL FFTD
      NK=1
      NC=5
      CALL INRECO
      NA=NA+NN
      NC=6
      CALL INRECO
      RETURN
      END
```

DLINE

```
      SUBROUTINE DLINE(A,B,NX,NY,K,L)
      COMPLEX A(NX,NY),B(NY)
      DO 1 J=1,NY
      IF(L.EQ.1) GOTO2
      B(J)=A(K,J)
      GOTO1
    2 A(K,J)=B(J)
    1 CONTINUE
      RETURN
      END
```

FTFB

```
      SUBROUTINE FTFB(F,NT,NB,W,TO,TM)
      COMPLEX F(NT),W(NT)
      COMMON FICT(392),N,I1(3),C(1)
      N=NT
      TT=TM-TO
      DT=TT/FLOAT(N)
      N1=N/2
      DO 1 I=1,N
      TX=TO+(I-1)*DT
      F(I)=F(I)*EXP(2.*TX)
      C(2*I-1)=REAL(F(I))
    1 C(2*I)=AIMAG(F(I))
      CALL DQFT1
      DO 2 I=1,N
    2 F(I)=CMPLX(C(2*I-1),C(2*I))
      DO 3 I=1,N1
      I1=I+N1
      F(I)=F(I)*W(I1)
    3 F(I1)=F(I1)*W(I)
      DO 4 I=1,N
      C(2*I-1)=REAL(F(I))
    4 C(2*I)=AIMAG(F(I))
      CALL DQFT1
      DO 5 I=1,N
    5 F(I)=CMPLX(C(2*I-1),C(2*I))/FLOAT(N)
      RETURN
      END
```

FILT

```
      SUBROUTINE FILT(W,NT,NB,TT)
      COMPLEX W(NT),Z,Z1
      AO=ALOG(2.)
      PI=2.*3.14159265
      N=NT
      N1=N/2
      NO=MOD(NB,2)
      M=(NB-NO)/2-1
      MO=1-NO
      DO 1 I=1,N
      OM=0.5*PI*(I-1-N1)/TT
      Z1=CMPLX(1.,OM)
      IF(NO.NE.1)GOTO2
      Z1=CMPLX(0.5,OM)
    2 CONTINUE
      CALL GAMMA(Z1)
      W(I)=Z1
      IF(NB.LE.2)GOTO3
      DO 4 J=MO,M
      XN=FLOAT(J)+FLOAT(NO)/2.
      Z=CMPLX(XN,OM)
    4 W(I)=W(I)*Z
    3 RR=COS(2.*OM*AO)
      RI=SIN(2.*OM*AO)
      Z=CMPLX(RR,RI)
      Z1=CMPLX(FLOAT(NB),-2.*OM)
      Z=Z/Z1
      RR=REAL(W(I))
      RI=-AIMAG(W(I))
      Z1=CMPLX(RR,RI)
      W(I)=W(I)*Z/Z1
      W(I)=CMPLX(REAL(W(I)),-AIMAG(W(I)))
    1 CONTINUE
      RETURN
      END
```

BESF

```
      SUBROUTINE BESF(BF,TF,NF)
      COMPLEX BF(NF)
      COMMON FICT(395),N,II(3),C(1)
C     N-ЧИСЛО ТОЧЕК СЕТКИ НА ОКРУЖНОСТИ
C              (NF<N/2)
      N=4096
      PI=2.*3.14159265
      D=PI/FLOAT(N)
      DO 1 I=1,N
      TX=(I-1)*D
      RR=COS(TF*SIN(TX))
      RI=-SIN(TF*SIN(TX))
      C(2*I-1)=RR
    1 C(2*I)=RI
      CALL DQFT1
      DO 2 I=1,NF
    2 BF(I)=CMPLX(C(2*I-1),C(2*I))/FLOAT(N)
      RETURN
      END
```

GAMMA1

```
      SUBROUTINE GAMMA1(G,NT,TO,TM,AB)
      COMPLEX G(NT),A
      COMMON FICT(398),N,II(3),C(1)
      N=NT
      N1=N/2
      DT=(TM-TO)/FLOAT(N)
      DO 1 I=1,N
      TX=TO+(I-1)*DT
      R=AB*TX-EXP(TX)
      R1=ABS(R)
      IF(R1.LT.169.) GOTO2
      G(I)=0.
      GOTO1
    2 G(I)=(TM-TO)*EXP(R)
    1 CONTINUE
      CALL TOCOM(G,N,1,1)
      CALL DQFT1
      CALL OUTCOM(G,N,1,1)
      DO 3 I=1,N1
      I1=I+N1
      A=G(I)
      G(I)=G(I1)
    3 G(I1)=A
      RETURN
      END
```

GAMMA

```
      SUBROUTINE GAMMA(Z)
      COMPLEX Z,S
C     N-ЧИСЛО СОМНОЖИТЕЛЕЙ В ПРОИЗВЕДЕНИИ
      N=1000
      RK=ALOG(FLOAT(N))
      R=REAL(Z)
      RA=AIMAG(Z)
      RR=COS(RK*RA)
      RI=SIN(RK*RA)
      S=CMPLX(RR,RI)
      S=S*EXP(RK*R)/Z
      DO 1 I=1,N
    1 S=S*FLOAT(I)/(Z+FLOAT(I))
      Z=S
      RETURN
      END
```

CLMN

```
      SUBROUTINE CLMN(A,B,NX,NY,K,L)
      COMPLEX A(NX,NY),B(NX)
      DO 1 I=1,NX
      IF(L.EQ.1) GOTO2
      B(I)=A(I,K)
      GOTO1
    2 A(I,K)=B(I)
    1 CONTINUE
      RETURN
      END
```

AMULT

```
      SUBROUTINE AMULT(A,B,S,NX,NY)
      COMPLEX A(NX,NY),B(NX,NY),S(NX,NY)
      DO 1 J=1,NY
      DO 1 I=1,NX
    1 S(I,J)=A(I,J)*B(I,J)
      RETURN
      END
```

TOCOM

```
      SUBROUTINE TOCOM(A,NX,NY,K)
      COMPLEX A(NX,NY)
      COMMON PICT(399),C(1)
      DO 1 J=1,NY
      DO 1 I=1,NX
      L=K+2*(I-1)+2*(J-1)*NY
      L1=L+1
      C(L)=REAL(A(I,J))
    1 C(L1)=AIMAG(A(I,J))
      RETURN
      END
```

OUTCOM

```
      SUBROUTINE OUTCOM(A,NX,NY,K)
      COMPLEX A(NX,NY)
      COMMON PICT(399),C(1)
      DO 1 J=1,NY
      DO 1 I=1,NX
      L=K+2*(I-1)+2*(J-1)*NY
      L1=L+1
    1 A(I,J)=CMPLX(C(L),C(L1))
      RETURN
      END
```

TRNSP

```
      SUBROUTINE TRNSP(A,B,NX,NY)
      COMPLEX A(NX,NY),B(NX,NY)
      DO 1 J=1,NY
      DO 1 I=1,NX
    1 B(I,J)=A(J,I)
      RETURN
      END
```

CONJG

```
      SUBROUTINE CONJG(A,B,NX,NY)
      COMPLEX A(NX,NY),B(NX,NY)
      DO 1 J=1,NY
      DO 1 I=1,NX
      R=REAL(A(I,J))
      X=AIMAG(A(I,J))
    1 B(I,J)=CMPLX(R,-X)
      RETURN
      END
```

ORIGINAL PAGE IS
OF POOR QUALITY

POULR

```
      SUBROUTINE POULR(A,HX,HY,NX,NY,B,NT,NF,RO,RM)
      COMPLEX A(NX,NY),B(NT,NF)
      X0=0.5*(NX-1)*HX
      Y0=0.5*(NY-1)*HY
      DT=ALOG(RM/RO)/FLOAT(NT)
      DO 1 L=1,NF
      DO 1 K=1,NT
    1 B(K,L)=0.
      DO 2 L=1,NF
      DO 2 K=1,NT
      RK=RO*EXP((K-1)*DT)
      FI=2.*3.14159265*(L-1)/FLOAT(NF)
      X=RK*COS(FI)
      Y=RK*SIN(FI)
      I=1+INT((X+X0)/HX)
      J=1+INT((Y+Y0)/HY)
      I1=I+1
      J1=J+1
      IF(I.LT.1) GOTO2
      IF(J.LT.1) GOTO2
      IF(I1.GT.NX) GOTO2
      IF(J1.GT.GY) GOTO2
      B(K,L)=0.25*(A(I,J)+A(I1,J)+A(I,J1)+A(I1,J1))
    2 CONTINUE
      RETURN
      END
```

REVOL

```
      SUBROUTINE REVOL(F,NO,N1,V,NX,NY,AL)
      COMPLEX F(NO,N1),V(NX,NY)
      N11=N1/2
      NO1=NO/2
      NX1=NX/2
      NY1=NY/2
      R1=COS(AL)
      R2=SIN(AL)
      DO 1 J=1,NY
      DO 1 I=1,NX
      I1=I-1-NX1
      J1=J-1-NY1
      X1=I1*R1+J1*R2
      Y1=-I1*R2+J1*R1
      I1=INT(X1+NO1)+1
      J1=INT(Y1+N11)+1
    1 V(I,J)=0.25*(F(I1,J1)+F(I1+1,J1)+F(I1,J1+1)+F(I1+1,J1+1))
      RETURN
      END
```

CORF

```
      SUBROUTINE CORF(F,V,S,NX,NY)
      COMPLEX F(NX,NY),V(NX,NY),S(NX,NY)
      COMMON FICT(395),N,NN,IB(2),C(1)
      N=NX
      NN=N*NY
      CALL TOCOM(F,N,NY,1)
      CALL RQFT2
      CALL OUTCOM(F,N,NY,1)
      CALL TOCOM(V,N,NY,1)
      CALL RQFT2
      CALL OUTCOM(V,N,NY,1)
      CALL AMULT(F,V,S,N,NY)
      CALL TOCOM(S,N,NY,1)
      CALL RQFT2
      CALL OUTCOM(S,N,NY,1)
      RETURN
      END
```